Segmentation Transformer

유 현 우
Vision and Display System Lab.
Sogang University

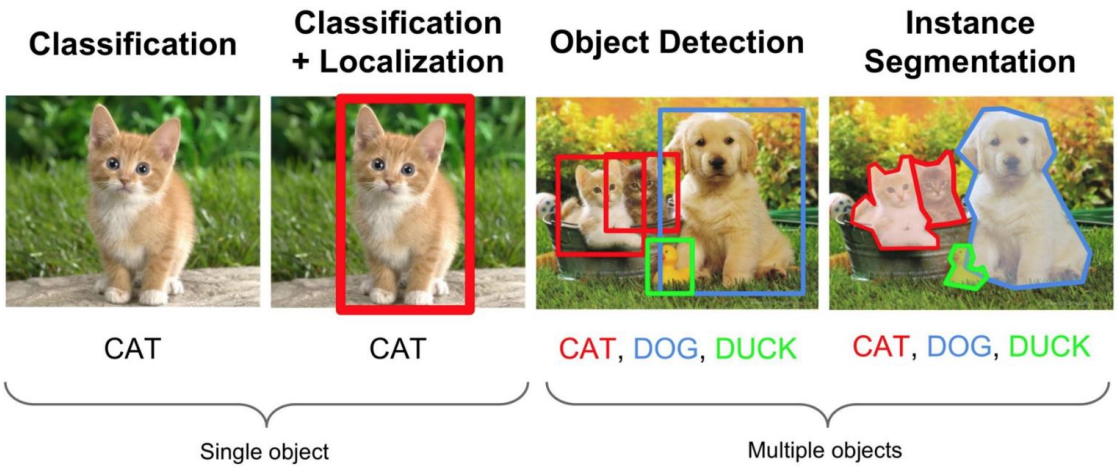
Outline

- Background
  - Semantic segmentation
  - Fully convolutional network (FCN)
- Segmentation Transformer
  - Semantic segmentation에서 FCN architecture의 한계
  - Proposed method (SETR)
- Experiment
- Conclusion

"Rethinking Semantic Segmentation from a Sequence-to-Sequence Perspective with Transformers" (CVPR 2021)

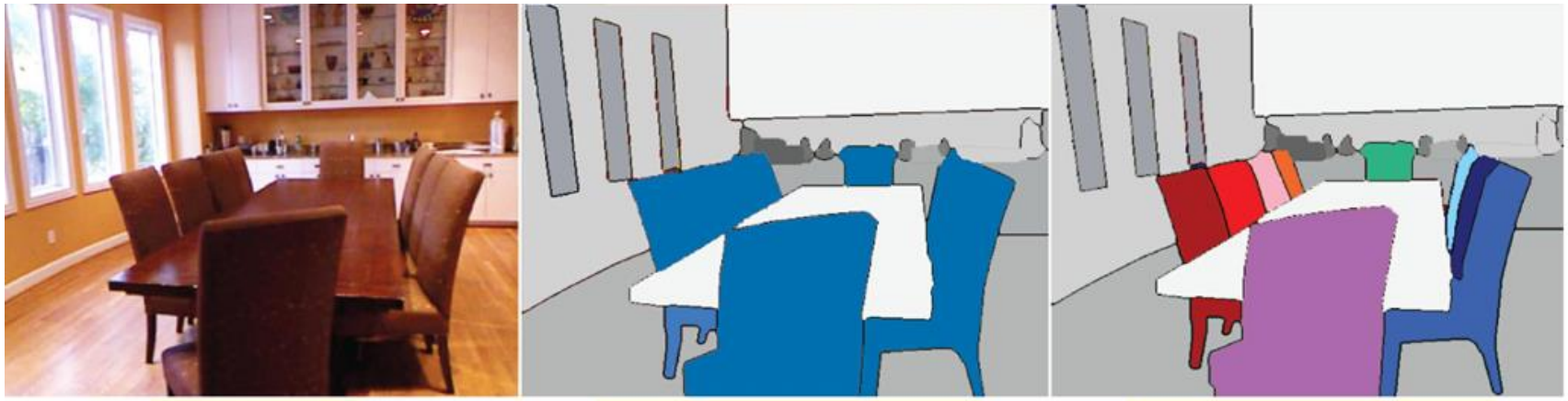
Semantic segmentation

- Classification
  - Image 전체 영역에 대해 classification 수행
- Localization + Detection
  - 공간정보를 이용해 object의 위치 파악
    - Bounding box의 형태로 object의 위치 파악
- Segmentation
  - 모든 픽셀에 대해 classification 수행



Semantic segmentation

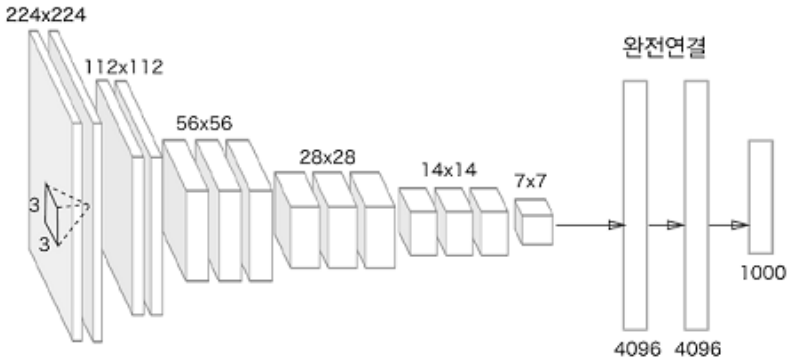
- Semantic segmentation
  - 각각의 pixel들이 어떤 class에 속하는지 구분
- Instance segmentation
  - 픽셀별로 속하는 class의 instance 또한 구분



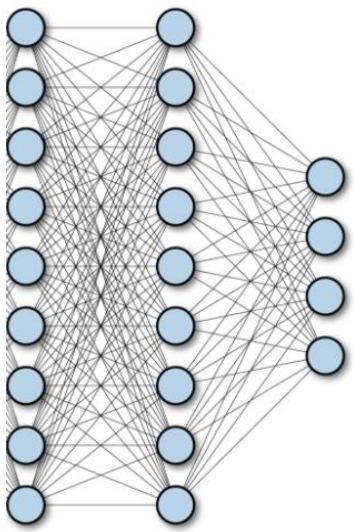
Input Image | Semantic Segmentation | Instance Segmentation

Fully convolutional network

- Image classification의 한계
  - 마지막 layer에 classifier로 fully connected layer를 이용하므로 공간정보 손실
    - Fully connected layer에서 모든 노드들이 weight와 곱해진 뒤 더해짐



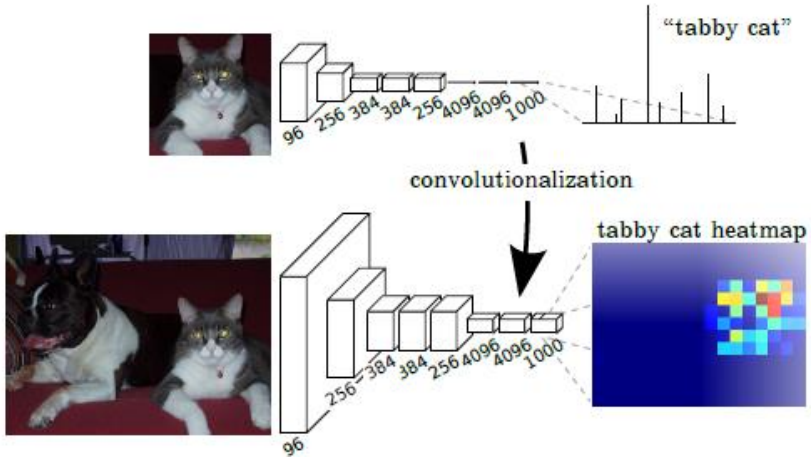
VGG model architecture



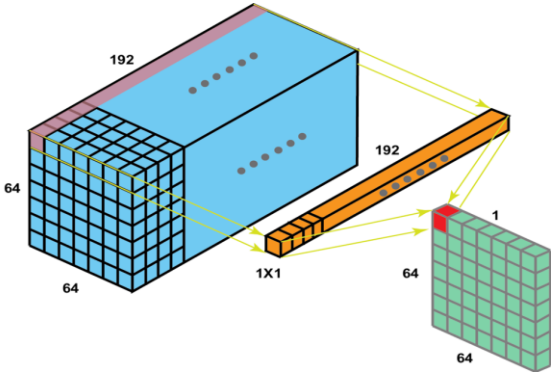
Fully connected layer 모식도

Fully convolutional network

- Fully Convolutional Network(FCN)
  - Classification을 위해 고안된 VGG model을 segmentation에 적용
  - Classifier로 fully connected layer 대신 1x1 convolution 사용
    - 인풋에 대해 channel축으로 class에 따른 feature 정보를 얻음
      - Heatmap은 픽셀별로 특정 class에 분류될 확률에 해당하는 수치를 포함
    - Spatial축에는 input image의 공간정보가 담김



VGG – 16 encoding

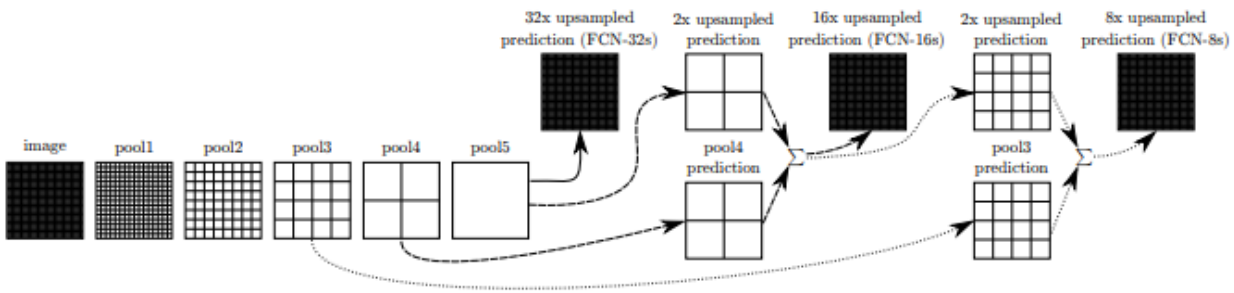


1×1 convolution 연산 모식도

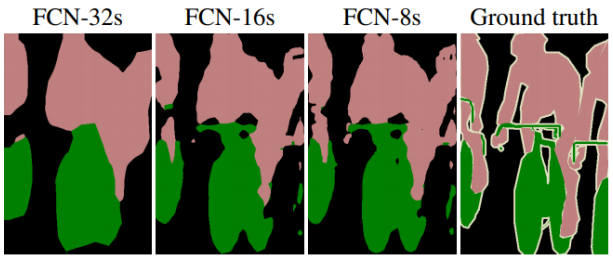
"Fully Convolutional Networks for Semantic Segmentation"(CVPR2015)

Fully convolutional network

- Upsampling
  - Bilinear interpolation 또는 Transposed convolution을 이용하여 upsampling
  - CNN layer의 pooling을 지나며 receptive field가 확장
  - Segmentation의 성능을 높이기 위해 다양한 크기의 receptive field를 갖는 feature map 활용
    - Receptive field가 작은 low level feature는 localization 정보를 포함
    - 다양한 크기의 receptive field를 갖는 feature를 이용하여 segmentation의 detail이 개선



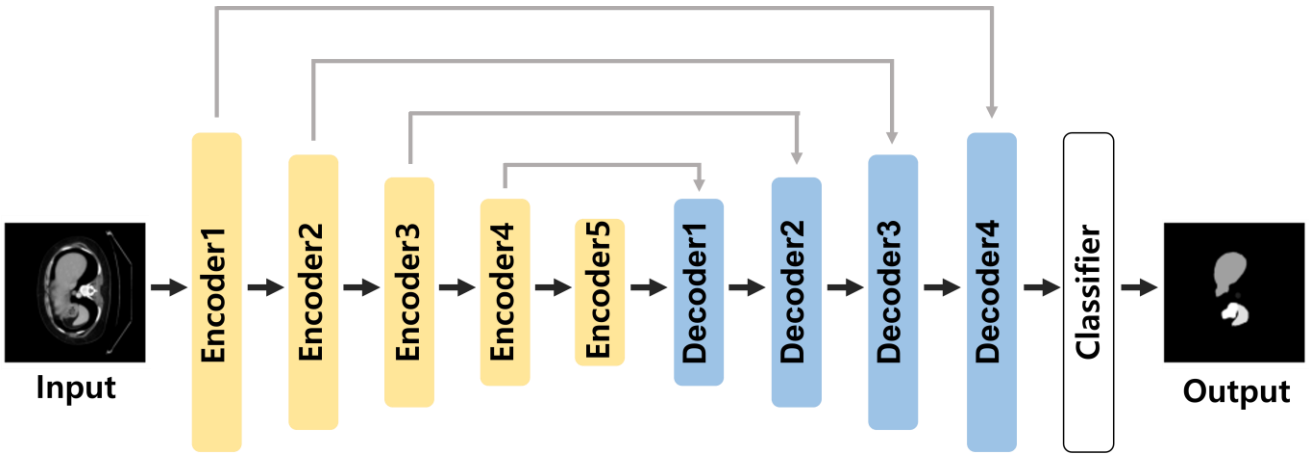
Upsampling 시 skip architecture



Skip archtiecute에 따른 결과영상

FCN architecture의 한계

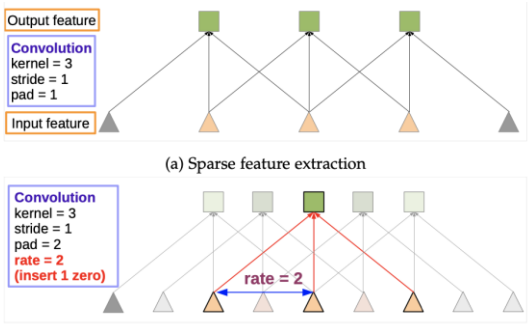
- Context modeling과 spatial resolution간에 trade-off 관계
  - Context modeling을 위해 stacked CNN layer를 거치며 receptive field를 확장
  - Convolution layer에서 pooling으로 인해 spatial resolution 감소



U-Net architecture

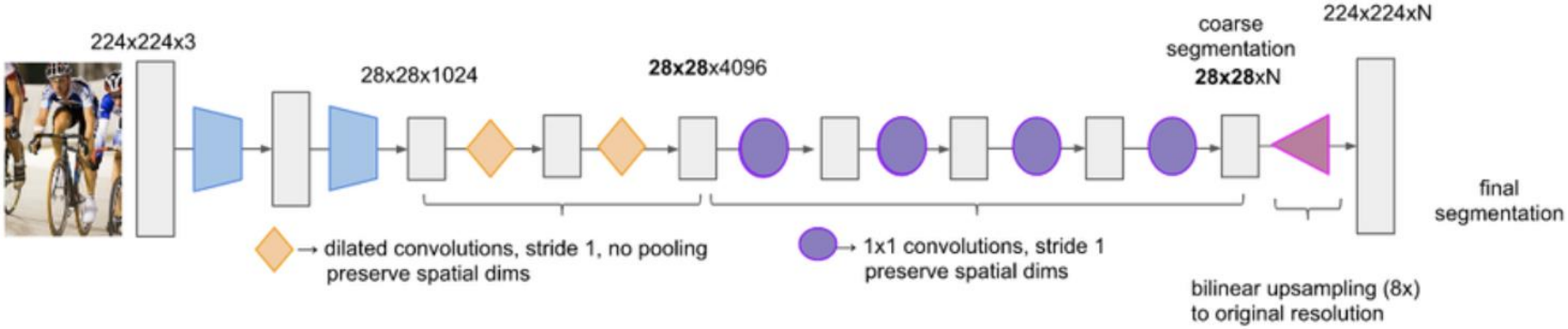
FCN based semantic segmentation

- Convolution operation을 개선
  - Deeplab series의 atrous convolution
    - Spatial resolution 감소 없이 receptive field를 확장시키는 효과
    - 기존 CNN layer와 결합하여 사용
    - Global context modeling을 위해 여러 층을 쌓아야 함



(a) Sparse feature extraction

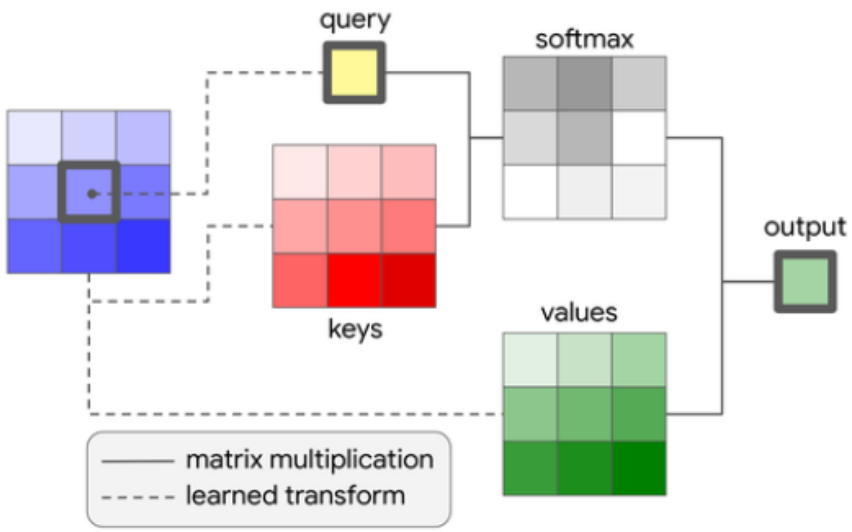
(b) Dense feature extraction



"SEMANTIC IMAGE SEGMENTATION WITH DEEP CON- VOLUTIONAL NETS AND FULLY CONNECTED CRFS" (ICLR2015)

FCN based semantic segmentation

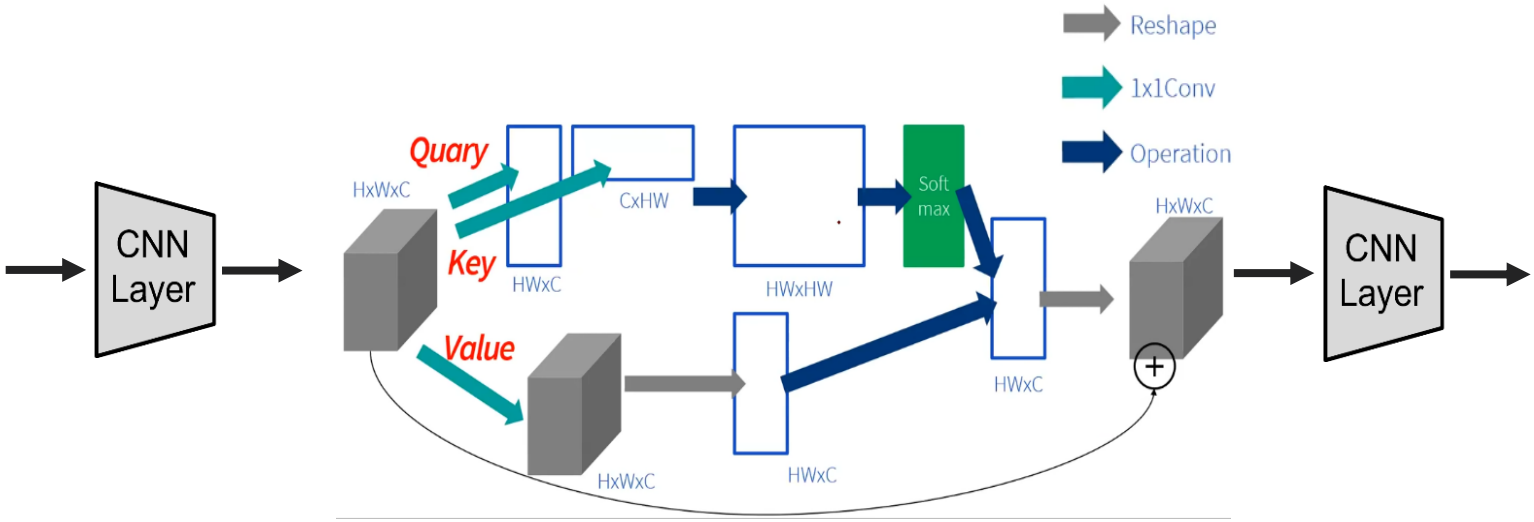
- Combination of FCN with attention module
  - Self attention mechanism
    - Input data로부터 query, key, value를 생성
    - Query와 key간의 유사도를 value에 적용시키는 방향으로 학습
    - Input data에 대해 global하게 attention을 수행할 수 있음



Attention mechanism

FCN based semantic segmentation

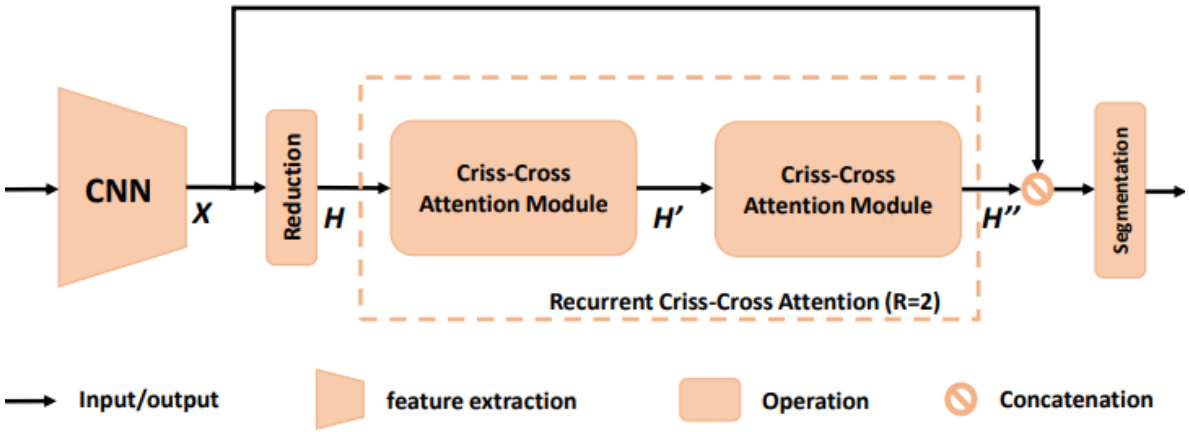
- Combination of FCN with attention module
  - Non-local Neural Networks
    - CNN layer와 결합되어 FCN의 구조적 한계를 벗어나지 못함
    - Self attention 과정 중에 모든 pixel에 대해 attention을 수행하므로 computation cost 문제 발생



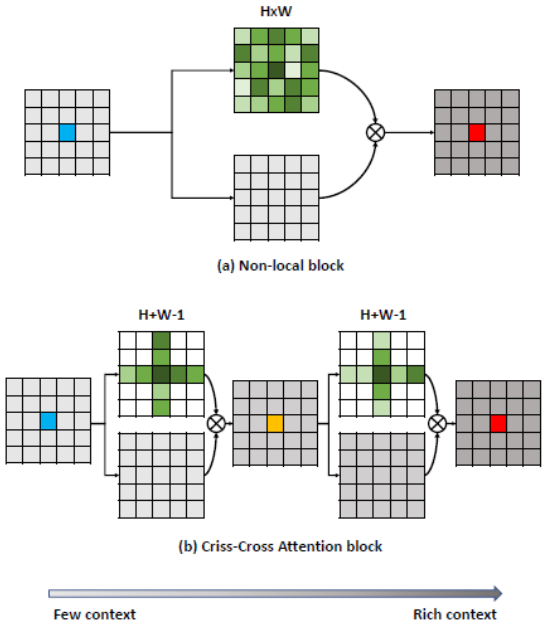
"Non-local Neural Networks" (CVPR2018)

FCN based semantic segmentation

- Combination of FCN with attention module
  - CCNet : Criss-Cross Attention for Semantic Segmentation
    - Non-local neural network 보다 computation cost가 개선
    - CNN layer와 결합되어 FCN의 구조적 한계를 벗어나지 못함



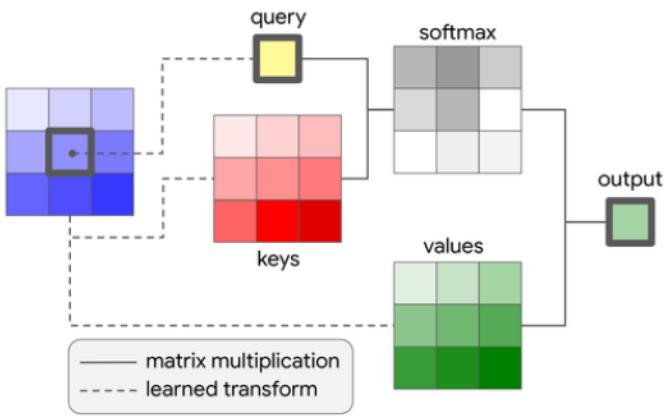
CCNet architecture



"CCNet : Criss-Cross Attention for Semantic Segmentation" (ICCV2019)

FCN based semantic segmentation

- Attention alone model
  - Stand-Alone Self-Attention in Vision Models
    - Local attention을 이용해 convolution 연산 대체
- CNN이 없어도 FCN의 구조적 특성이 변하지 않음
  - Attention 기법을 이용해서 spatial resolution을 downsampling하여 context modeling

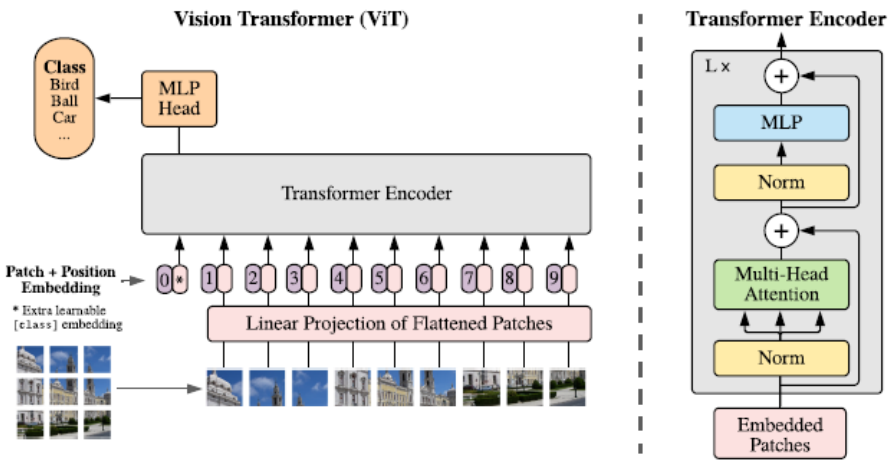


Stand-alone self attention의 local attention layer

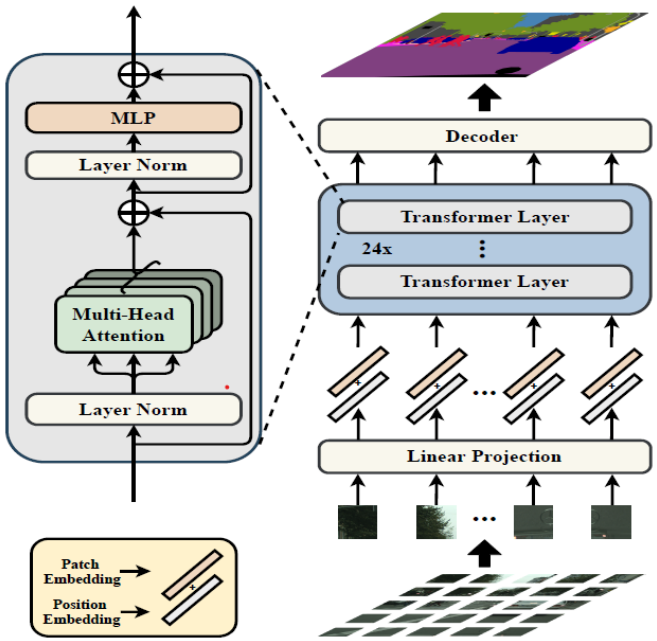
"Stand-Alone Self-Attention in Vision Models" (NIPS 2019)

Segmentation Transformer(SETR)

- Contribution
  - SETR은 stack형 convolution layer기반 encoder를 transformer로 대체
  - **Spatial resolution의 감소 없이 global context modeling을 수행**



Vision transformer (ViT) model overview

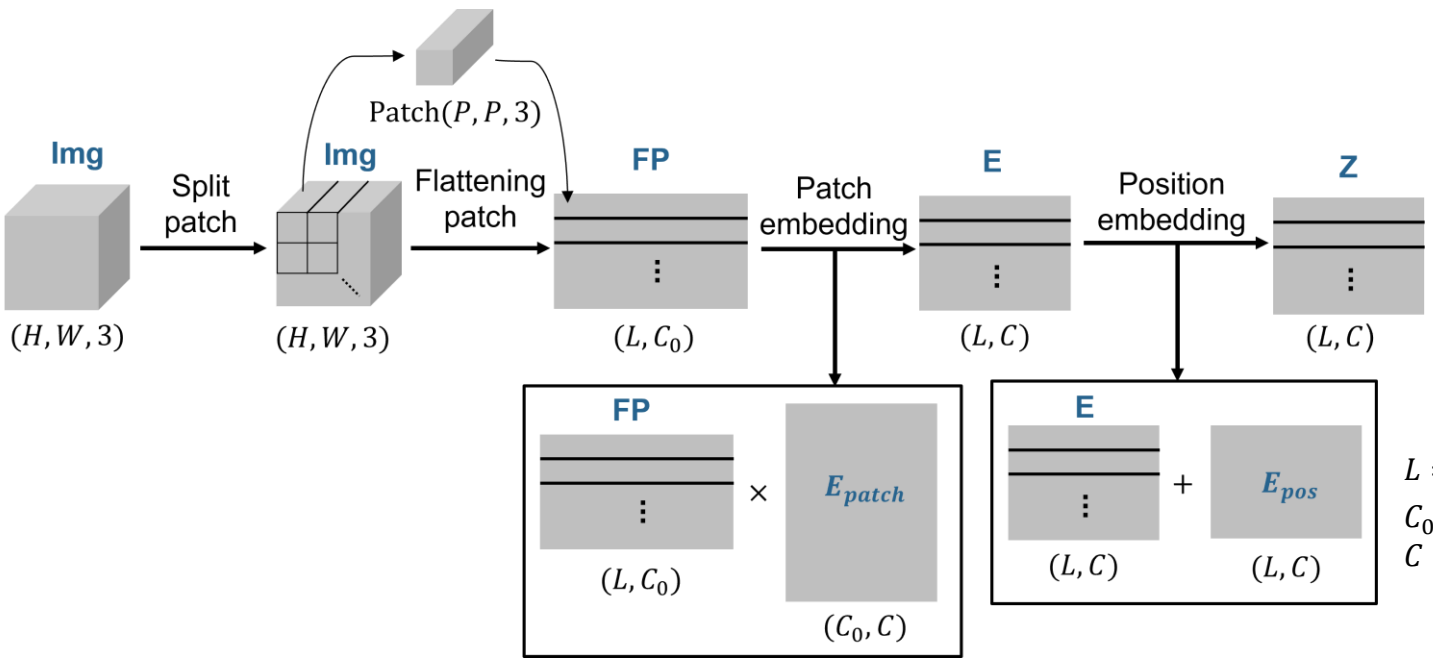


Segmentation transformer (SETR) model overview

"An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale" (ICLR2021)

Segmentation Transformer(SETR)

- Embedding
  - Patch embedding에서 linear projection을 수행하여 1D sequence 를 얻음
  - 입력될 patch들 사이에 공간정보가 없으므로 position embedding을 수행



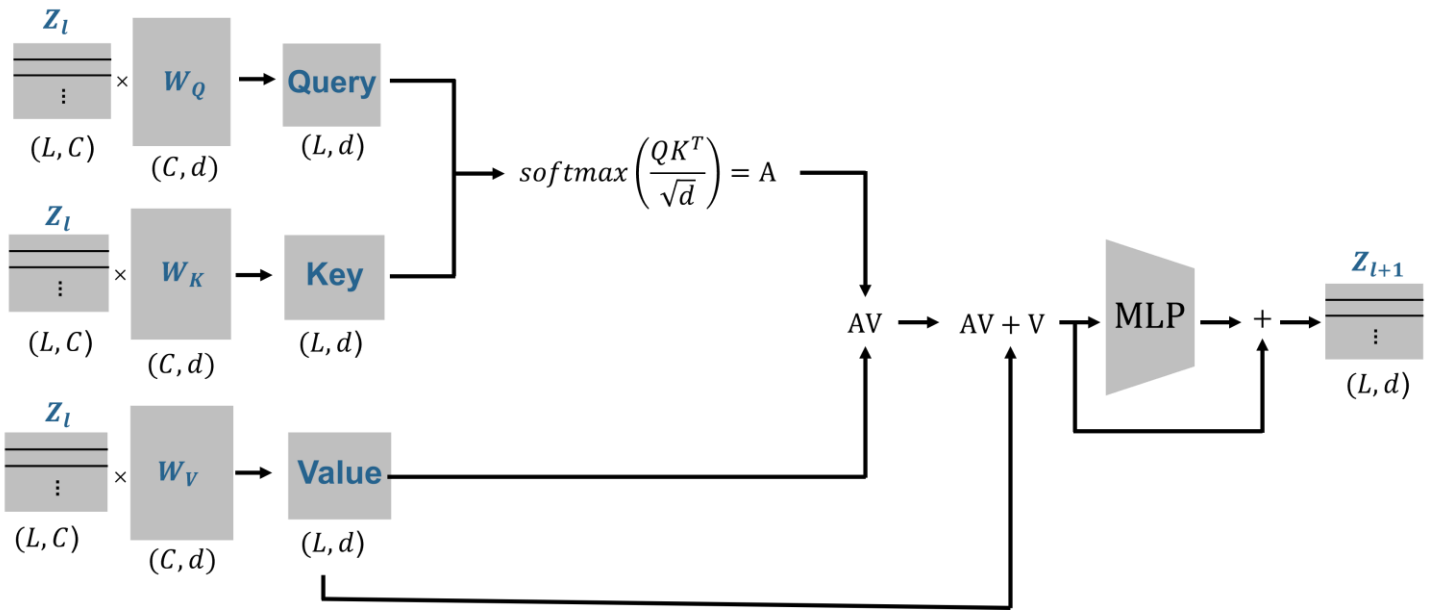
$L = \frac{H}{P} * \frac{W}{P}$: *patch의 개수*

$C_0 = P * P * C$: *patch 픽셀 수*

C : *hidden channel size*

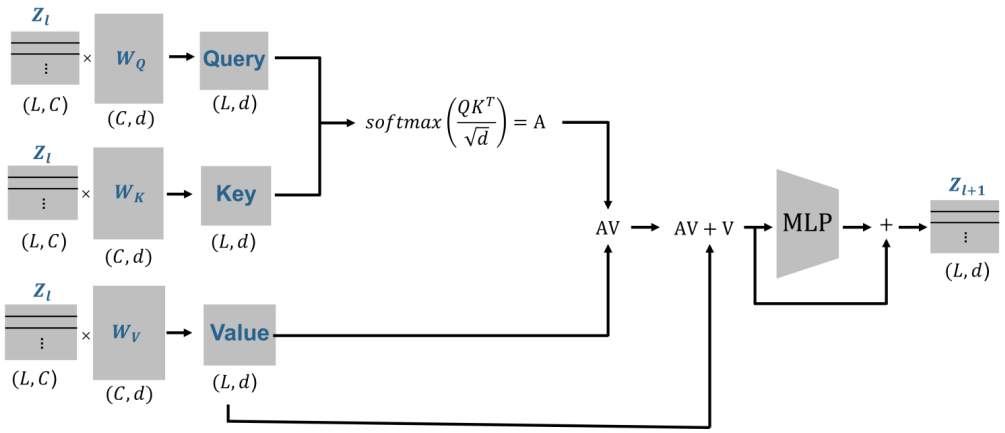
Segmentation Transformer(SETR)

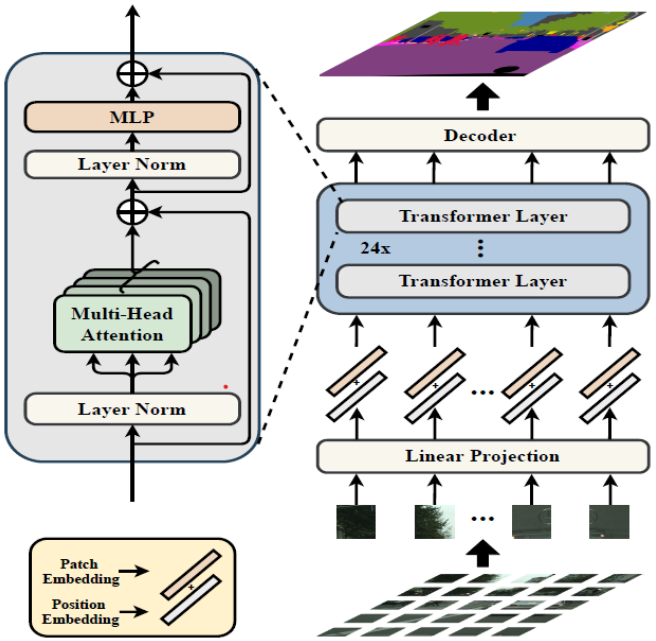
- Encoding
  - $SA(Z_l) = Z_l + softmax(\frac{Z_l W_Q (Z_l W_K)^T}{\sqrt{d}})\,(Z_l W_V)$ (SA : self attention)
  - Query와 Key를 이용해 유사도를 구한 뒤 이를 Value에 적용
  - Patch의 개수 즉, sequence length인 L의 크기가 입력과 출력에서 동일



Segmentation Transformer(SETR)

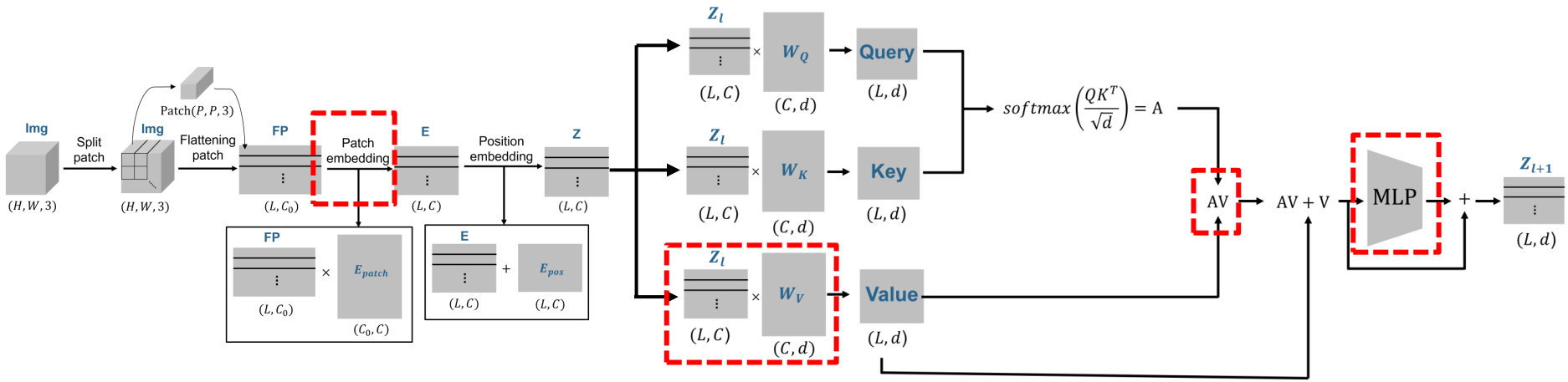
- Encoding
  - $Z_{l+1} = MSA(Z_l) + MLP(MSA(Z_l))$
  - MSA(multi-head self attention)은 self attention의 과정을 병렬적으로 여러 개 처리
  - MLP(multi-layer perceptron)
  - Feature representation = $\{Z_1, Z_2, \cdots, Z_L\}$

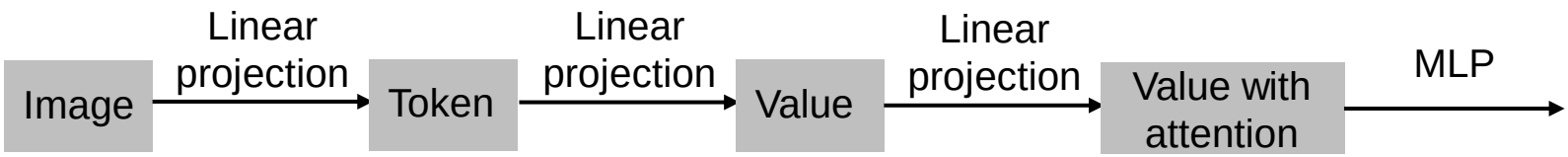




Segmentation Transformer(SETR)

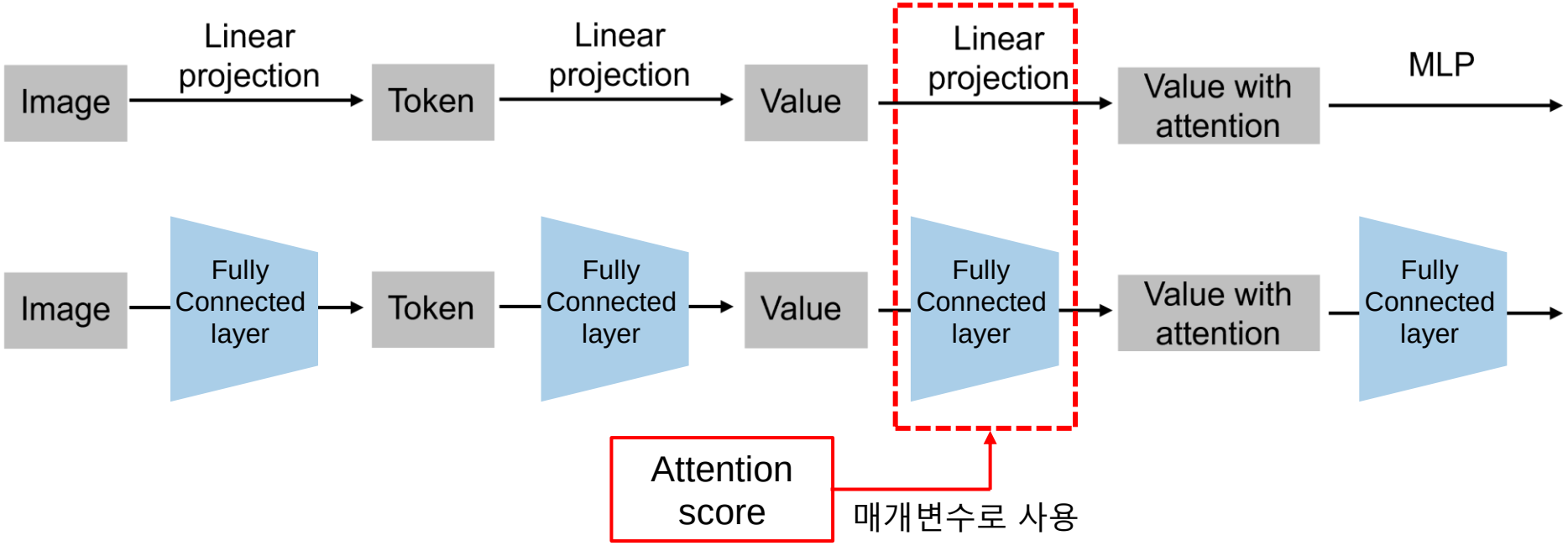
- Encoding





Segmentation Transformer(SETR)

- Encoding
  - Linear projection은 fully connected layer와 동일한 연산
  - Fully connected layer로 이루어진 network로 생각해볼 수 있음
  - Query, Key의 유사도를 Value에 적용하여 feature를 추출하는 방향으로 학습

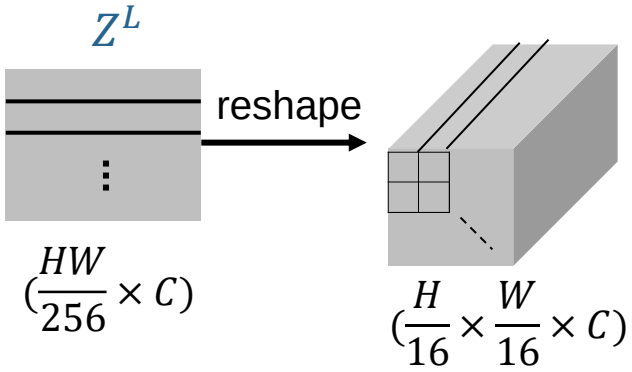


Segmentation Transformer(SETR)

- Decoding
  - Reshape
    - Patch size p = 16으로 하여 feature representation Z^L 의 shape이 $\left(\frac{HW}{256}, C\right)$이 되도록 설정
    - Z^L을 $\frac{H}{16} \times \frac{W}{16} \times C$으로 reshape

Model	T-layers	Hidden size	Att head
T-Base	12	768	12
T-Large	24	1024	16

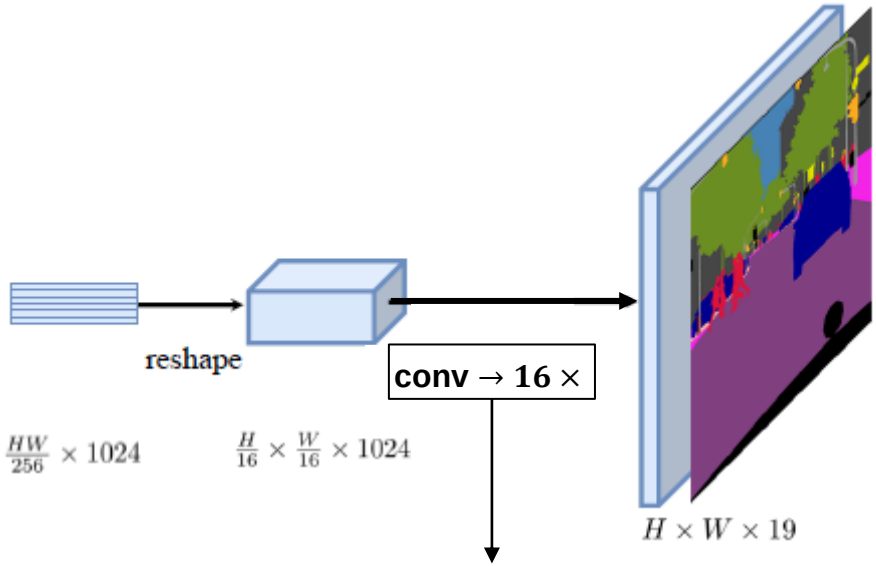
Table 1. Configuration of Transformer backbone variants.



C : *hidden size*

Segmentation Transformer(SETR)

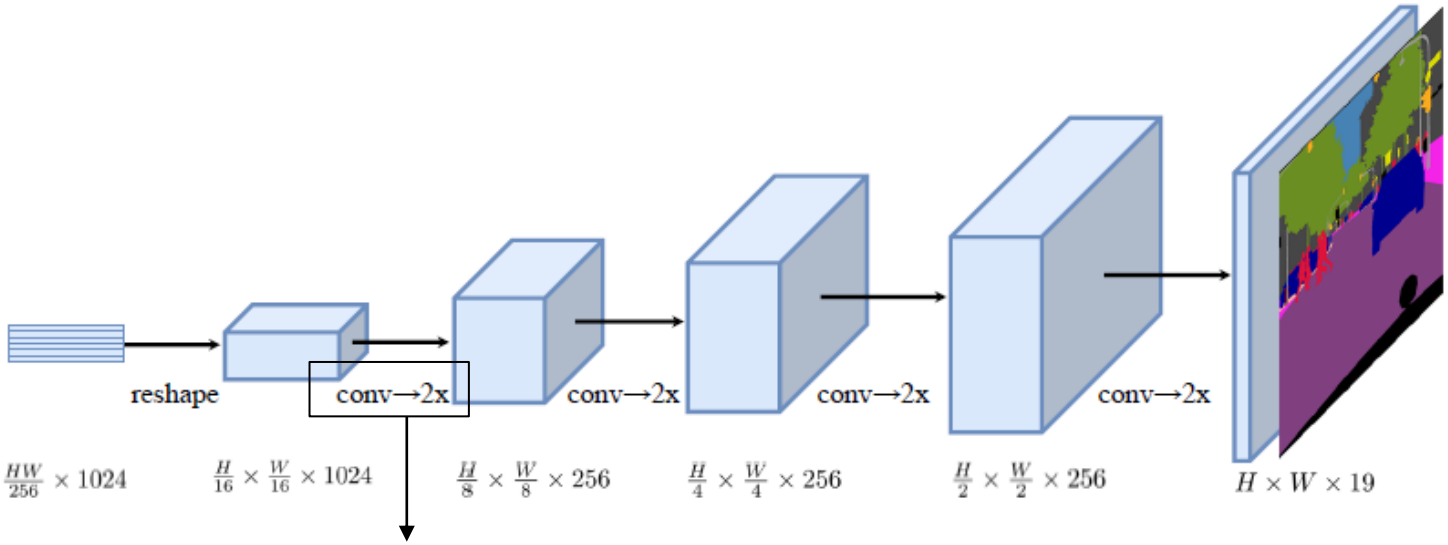
- Decoding
  - Naïve upsampling
    - Conv : 1×1 conv + sync batch norm (w/Relu) + 1×1 conv
    - Bilinear upsampling : 16× Bilinear interpolation



1×1 conv → bilinear upsampling

Segmentation Transformer(SETR)

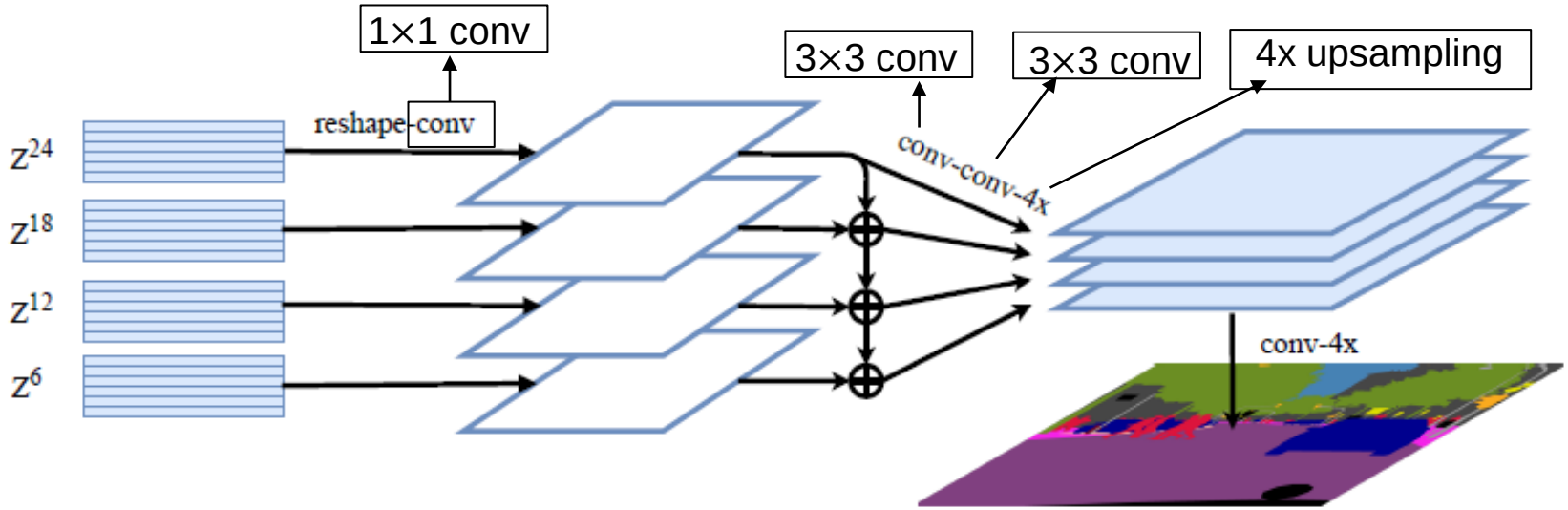
- Decoding
  - Progressive upsampling (PUP)
    - Conv : 1×1 conv
    - Bilinear upsampling : 2× Bilinear interpolation



1×1 conv → bilinear upsampling

Segmentation Transformer(SETR)

- Decoding
  - Multi-level feature aggregation (MLA)
    - 6,12,18,24번째 transformer layer에서 나온 feature representation을 skip connection하여 사용
    - 각각의 feature representation에 대해 1× 1 conv를 진행
    - 3×3 conv를 두 번 사용한 후 4× bilinear interpolation을 진행
    - Skip connection 된 feature representation과 concat하여 4× bilinear interpolation을 진행



Segmentation Transformer(SETR)

- Experiment

Method	Pre	Backbone	#Params	40k	80k
FCN [39]	1K	R-101	68.59M	73.93	75.52
Semantic FPN [39]	1K	R-101	47.51M	-	75.80
Hybrid-Base	R	T-Base	112.59M	74.48	77.36
Hybrid-Base	21K	T-Base	112.59M	76.76	76.57
Hybrid-DeiT	21K	T-Base	112.59M	77.42	78.28
SETR-*Naïve*	21K	T-Large	305.67M	77.37	77.90
SETR-*MLA*	21K	T-Large	310.57M	76.65	77.24
SETR-*PUP*	21K	T-Large	318.31M	78.39	79.34
SETR-*PUP*	R	T-Large	318.31M	42.27	-
SETR-*Naïve-Base*	21K	T-Base	87.69M	75.54	76.25
SETR-*MLA-Base*	21K	T-Base	92.59M	75.60	76.87
SETR-*PUP-Base*	21K	T-Base	97.64M	76.71	78.02
SETR-*Naïve-DeiT*	1K	T-Base	87.69M	77.85	78.66
SETR-*MLA-DeiT*	1K	T-Base	92.59M	78.04	78.98
SETR-*PUP-DeiT*	1K	T-Base	97.64M	**78.79**	**79.45**

Table 2. **Comparing SETR variants** on different pre-training strategies and backbones. All experiments are trained on Cityscapes train fine set with batch size 8, and evaluated using the single scale test protocol on the Cityscapes validation set in mean IoU (%) rate. "Pre" denotes the pre-training of transformer part. "R" means the transformer part is randomly initialized.

Method	Pre	Backbone	ADE20K	Cityscapes
FCN [39]	1K	R-101	39.91	73.93
FCN	21K	R-101	42.17	76.38
SETR-*MLA*	21K	T-Large	**48.64**	76.65
SETR-*PUP*	21K	T-Large	48.58	78.39
SETR-*MLA-DeiT*	1K	T-Large	46.15	78.98
SETR-*PUP-DeiT*	1K	T-Large	46.24	**79.45**

Table 3. **Comparison to FCN with different pre-training** with single-scale inference on the ADE20K val and Cityscapes val set.

Method	Backbone	mIoU
PSPNet [60]	ResNet-101	78.40
DenseASPP [50]	DenseNet-161	80.60
BiSeNet [52]	ResNet-101	78.90
PSANet [61]	ResNet-101	80.10
DANet [18]	ResNet-101	81.50
OCNet [55]	ResNet-101	80.10
CCNet [25]	ResNet-101	81.90
Axial-DeepLab-L [47]	Axial-ResNet-L	79.50
Axial-DeepLab-XL [47]	Axial-ResNet-XL	79.90
SETR-*PUP* (100k)	T-Large	81.08
SETR-*PUP*‡	T-Large	81.64

Table 7. **Comparison on the Cityscapes test set.** ‡: trained on fine and coarse annotated data.

Segmentation Transformer(SETR)

- Experiment

Method	Pre	Backbone	#Params	mIoU
FCN (160k, SS) [39]	1K	ResNet-101	68.59M	39.91
FCN (160k, MS) [39]	1K	ResNet-101	68.59M	41.40
CCNet [25]	1K	ResNet-101	-	45.22
Strip pooling [23]	1K	ResNet-101	-	45.60
DANet [18]	1K	ResNet-101	69.0M	45.30
OCRNet [54]	1K	ResNet-101	71.0M	45.70
UperNet [49]	1K	ResNet-101	86.0M	44.90
Deeplab V3+ [11]	1K	ResNet-101	63.0M	46.40
SETR-*Naïve* (160k, SS)	21K	T-Large	305.67M	48.06
SETR-*Naïve* (160k, MS)	21K	T-Large	305.67M	48.80
SETR-*PUP* (160k, SS)	21K	T-Large	318.31M	48.58
SETR-*PUP* (160k, MS)	21K	T-Large	318.31M	50.09
SETR-*MLA* (160k, SS)	21K	T-Large	310.57M	48.64
SETR-*MLA* (160k, MS)	21K	T-Large	310.57M	**50.28**
SETR-*PUP-DeiT* (160k, SS)	1K	T-Base	97.64M	46.34
SETR-*PUP-DeiT* (160k, MS)	1K	T-Base	97.64M	47.30
SETR-*MLA-DeiT* (160k, SS)	1K	T-Base	92.59M	46.15
SETR-*MLA-DeiT* (160k, MS)	1K	T-Base	92.59M	47.71

Table 4. **State-of-the-art comparison on the ADE20K dataset.** Performances of different model variants are reported. SS: Single-scale inference. MS: Multi-scale inference.

Method	Backbone	mIoU
FCN (40k, SS) [39]	ResNet-101	73.93
FCN (40k, MS) [39]	ResNet-101	75.14
FCN (80k, SS) [39]	ResNet-101	75.52
FCN (80k, MS) [39]	ResNet-101	76.61
PSPNet [60]	ResNet-101	78.50
DeepLab-v3 [10] (MS)	ResNet-101	79.30
NonLocal [48]	ResNet-101	79.10
CCNet [25]	ResNet-101	80.20
GCNet [4]	ResNet-101	78.10
Axial-DeepLab-XL [47] (MS)	Axial-ResNet-XL	81.10
Axial-DeepLab-L [47] (MS)	Axial-ResNet-L	81.50
SETR-*PUP* (40k, SS)	T-Large	78.39
SETR-*PUP* (40k, MS)	T-Large	81.57
SETR-*PUP* (80k, SS)	T-Large	79.34
SETR-*PUP* (80k, MS)	T-Large	**82.15**

Table 6. **State-of-the-art comparison on the Cityscapes validation set.** Performances of different training schedules (*e.g.*, 40k and 80k) are reported. SS: Single-scale inference. MS: Multi-scale inference.

Segmentation Transformer(SETR)

- Experiment

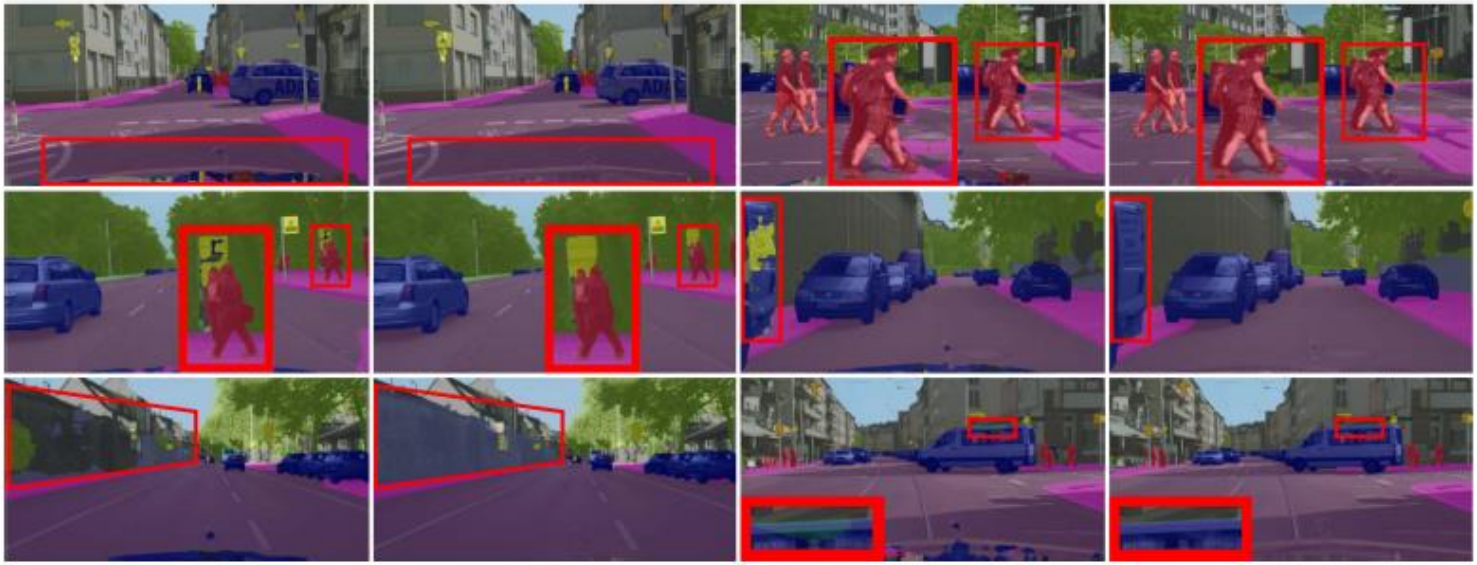


Figure 4. **Qualitative results on Cityscapes:** SETR (right column) vs. dilated FCN baseline (left column) in each pair. Best viewed in color and zoom in.